DOI: https://doi.org/10.32347/2707-501x.2023.52(3).217-226
УДК 004.2

А.А. Бугров
аспірант кафедри інформаційних технологій проектування та прикладної математики,
https://orcid.org/0000-0001-6986-1595
С.А. Теренчук
кандидат фізико-математичних наук,
доцент кафедри інформаційних технологій, професор
orcid.org/0000-0001-6527-4123
Київський національний університет будівництва і архітектури, м. Київ

ПРИНЦИПИ І МЕТОДИ АВТОМАТИЧНОГО МАСШТАБУВАННЯ ВИСОКОНАВАНТАЖЕНИХ СИСТЕМ

В роботі досліджено широке коло застосувань і основні підходи, які використовують для обробки інтернет-трафіку такі гіганти. як Netflix, YouTube, Disney+, TikTok, Facebook, Xbox Live, PlayStation Downloads і Amazon Prime. Показано, що визначальною особливістю високонавантажених систем є необхідність автоматизованого управління ресурсами при обробці великих обсягів запитів в режимі реального часу з забезпеченням високого рівня доступності, надійності і стабільності в умовах динамічно мінливих робочих навантажень. Проаналізовано вплив ключових метрик ефективності на інші характеристики функціонування системи. На основі цього аналізу систематизовано та структуровано набір ключових показників системи, що гарантує оперативність системи на зміни робочого навантаження в непередбачуваних умовах. Отримані результати свідчать про те, що пропускна здатність системи залежить від адаптивного балансування навантаження, тоді як масштабованість визначає здатність системи до адаптації. Таким чином ефективне управління ресурсами і алгоритми адаптивного масштабування надають можливість знизити витрати без втрат продуктивності. Також показано, що ефективність і стійкість високонавантажених систем суттєво залежить від архітектурних рішень. При чому вертикальне масштабування системи дає змогу підвищувати продуктивність окремих вузлів системи, а горизонтальне масштабування забезпечує кращу відмовостійкість. Встановлено, що основна проблема в досягненні економічної ефективності полягає в балансуванні між масштабованістю, продуктивністю та відмовостійкістю, а комбіновані стратегії масштабування можуть узгодити ці цілі Підкреслено актуальність розробки моделей і методів оптимізації часу регулювання навантаження та окреслено напрямки подальших досліджень. Результати цього дослідження в подальшому можуть бути використані для побудови адаптивних моделей і методів

управління ресурсами високонавантажених розподілених систем, що здатні ефективно функціонувати в умовах динамічних навантажень і зростаючих вимог до результативності керування ресурсами.

Ключові слова: адаптивність, висока продуктивність, динамічне масштабування, економічна ефективність, інформаційна система, інформаційна технологія, керування ресурсами, програмне забезпечення.

Вступ. За оцінками глобальної платформи даних і бізнес-аналітики Datareportal [1], кількість користувачів мережі інтернет останні роки характеризується стійким зростанням і на період січня 2022 р. інтернетом користується приблизно 62.5% населення планети.

Перманентно зростаючі обсяги інформації, що обробляється і зберігається в сучасних системах обробки потокових даних вимагають швидкого і точного розподілу ресурсів. Це означає, що ефективність і стабільність роботи сучасних високонавантажених систем залежить від їх здатності адаптуватися до динамічних і не завжди передбачуваних змін навантаження.

Наразі розробка і імплементація в високонавантажених систем в діяльність багатьох галузей народного господарства є важливим завданням, але унікальні особливості цих систем потребують індивідуального підходу до вибору метрик ефективності, масштабування і оптимізації архітектурних рішень, які забезпечать їм високу продуктивність, адаптивність та економічність. Саме тому дослідження впливу основних критеріїв надійності та ефективності на функціонування високонавантажених систем в умовах динамічних навантажень і можливість знизити витрати без втрат продуктивності є актуальним.

Огляд актуальних наукових досліджень. Дослідження різноманітних застосувань і промислових секторів, що варіюють від пошукових машин до соціальних мереж, в яких високонавантажені розподілені системи відіграють ключову роль, показав, що наразі відомі високонавантажені системи включають платформи потокового відео (Netflix, YouTube), великі соціальні мережі (Facebook, Twitter) і хмарні сервіси (Amazon Web Services, Microsoft Azure), які демонструють здатність обробляти величезні обсяги даних з високою надійністю і швидкістю (рис. 1).

Такі гіганти як Netflix, YouTube, Disney+, TikTok, Facebook, Xbox Live, PlayStation Downloads і Amazon Prime здатні обробляти більше 52,65 % всього інтернет-трафіку, що генерується лише 10 застосунками, завдяки комплексному впровадженню сучасних інформаційних технологій [2].

Основними підходами, які використовують ці компанії, є [3]:

1. Розподілені архітектури і мікросервісна організація.
2. Використання мережевих технологій і контент-доставки.

3. Оптимізація мережевих протоколів і застосування сучасних алгоритмів балансування навантаження.
4. Постійний моніторинг, аналіз і адаптивне масштабування.

APP TOTAL VOLUME 2022

	Application	Total Volume
1	Netflix	13.74%
2	YouTube	10.51%
3	Generic QUIC	5.41%
4	HTTP Media Stream	4.33%
5	Disney+	4.20%
6	Tik Tok	3.55%
7	Facebook	2.83%
8	Xbox Live	2.71%
9	Playstation Downloads	2.70%
10	Amazon Prime	2.67%

а

DOWNSTREAM TRAFFIC

	Application	Total Volume
1	Netflix	14.93%
2	YouTube	11.62%
3	Generic QUIC	5.88%
4	Disney+	4.49%
5	TikTok	3.93%
6	HTTP Media Stream	3.72%
7	Playstation Downloads	2.95%
8	Xbox Live	2.91%
9	Facebook	2.87%
10	Amazon Prime	2.83%

б

UPSTREAM TRAFFIC

	Application	Total Volume
1	Netflix	8.78%
2	HTTP Media Stream	6.89%
3	YouTube	5.90%
4	Generic Web Browsing	3.85%
5	HTTP download	3.70%
6	Generic QUIC	3.43%
7	Generic Messaging	3.17%
8	Disney+	2.98%
9	Hulu	2.79%
10	Facebook	2.67%

в

Рис.1. Рейтинг компаній за загальним об'ємом трафіку (а), об'ємом завантаження (б) та об'ємом вивантаження (в) [2]

Розподіл функціональності між незалежними сервісами системи створює передумови для досягнення її оптимальної продуктивності і адаптивності, що дозволяє значно покращити стабільність роботи застосунків, які функціонують в умовах змінних навантажень, у середовищі з високими вимогами до затримок і безперервності. Тому мікросервісна архітектура (рис. 2), що передбачає поділ застосунку на незалежні сервіси, є одним із основних підходів у проектуванні.

Економічна ефективність високонавантаженої системи залежить від балансу між споживанням обчислювальних ресурсів і продуктивністю. Тому оцінка економічної ефективності вимагає комплексного підходу, що потребує аналізу продуктивності, адаптивності, стійкості і економічної доцільності використання обчислювальних ресурсів. [4].

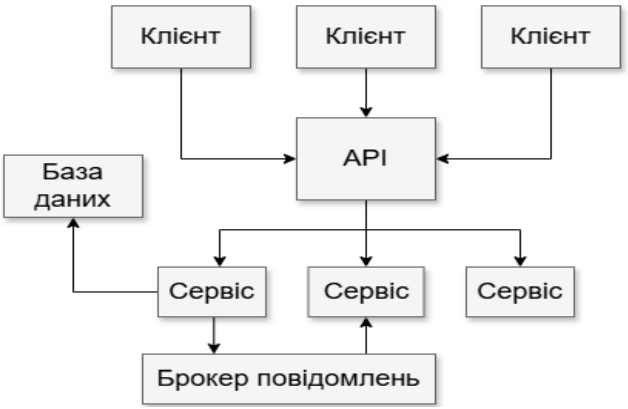

Рис.2. Приклад мікросервісної архітектури високонавантаженої системи

Метою статті є аналіз метрик ефективності, що відіграють ключову роль у забезпеченні стабільної і ефективної роботи високонавантажених розподілених систем та визначення механізму забезпечення комплексної результативності керування ресурсами.

Виклад основного матеріалу. Одним з ключових показників ефективності високонавантаженої системи є продуктивність. Метрики продуктивності охоплюють пропускну здатність, швидкість обробки запитів і середній час відгуку. Ці показники критичні для оцінки можливостей системи щодо роботи під навантаженням і впливають на кінцевий користувацький досвід [4].

Пропускна здатність визначає максимальну кількість запитів чи операцій, що виконуються системою за одиницю часу. Цей фактор безпосередньо впливає на здатність системи обробляти великі обсяги даних і забезпечувати високу продуктивність при значному навантаженні. Пропускна здатність є особливо критичною для мікросервісних архітектур і розподілених баз даних, оскільки їхня робота передбачає обробку великої кількості одночасних запитів, що можуть створювати значне навантаження на обчислювальні ресурси [5]. Якщо система не має ефективного механізму масштабування, то при збільшенні навантаження можливе перевантаження серверів, що призводить до зростання часу відгуку і втрати продуктивності.

Підхід до підтримки високої пропускної здатності в умовах змінного навантаження, що базується на адаптивному балансуванні навантаження, яке передбачає рівномірний розподіл запитів між доступними обчислювальними вузлами досліджено в [6]. Цей підхід дозволяє уникати ситуацій, коли одні вузли перевантажуються, тоді як інші недостатньо задіяні. Адаптивне балансування здійснюється за допомогою алгоритмів динамічного розподілу запитів. Така стратегія забезпечує оптимальне використання ресурсів, запобігає утворенню вузьких місць і сприяє підвищенню загальної ефективності системи.

Залежно від архітектури системи і характеру навантаження, пропускна здатності може покращуватися шляхом оптимізації механізмів кешування, партиціювання даних і паралельної обробки запитів. Використання таких методів масштабування, як реплікація даних або автоматичне горизонтальне масштабування, теж дозволяє збільшити пропускну здатність без значного зростання часу відгуку. Проте досягнення оптимального рівня пропускної здатності вимагає врахування низки параметрів, включаючи архітектурні обмеження, характеристики апаратного забезпечення та програмні механізми управління навантаженням.

Середній час відгуку визначає інтервал між моментом надсилання запиту користувачем або внутрішнім сервісом і отриманням відповіді від системи. Цей показник має вирішальне значення для інтерактивних сервісів, таких як веб-додатки, системи онлайн-банкінгу і потокові платформи, де навіть незначні затримки можуть суттєво впливати на користувацький досвід і рівень задоволеності [7].

Методи оптимізації середнього часу відгуку, включаючи кешування запитів, використання балансувальників навантаження і асинхронну обробку даних проаналізовані в [8]. В цій роботі показано, що застосування алгоритмів кешування на рівні серверів зменшує затримки обробки повторюваних запитів у середньому на 40%, що суттєво покращує продуктивність систем в сценаріях з високим рівнем однотипних запитів.

Використання асинхронної обробки в програмному забезпеченні дозволяє зменшити час очікування в системах, у яких операції введення-виведення можуть спричиняти суттєві затримки, особливо під час роботи з базами даних або зовнішніми API. Проте, середній час відгуку не завжди є достатньо адекватним показником оцінки продуктивності системи, оскільки він усереднює запити, не враховуючи явище хвостової латентності.

На відміну від середнього часу відгуку, який свідчить про загальну продуктивність системи, хвостові затримки фокусуються на найгірших сценаріях роботи окремих запитів, що можуть суттєво впливати на загальну якість обслуговування чи спричинити каскадну деградацію продуктивності [9]. Це особливо критично для телекомунікаційних і

фінансових систем, де гарантована швидкість обробки даних є ключовою вимогою до безперебійного надання послуг коли навіть поодинокі випадки надмірних затримок можуть спричинити серйозні наслідки.

Таким чином, для комплексної оцінки ефективності розподілених систем необхідно враховувати не лише середній час відгуку, але й показники хвостової латентності, що дозволяють точніше визначити стабільність роботи системи під змінними навантаженнями [10].

Відмовостійкість визначає здатність системи зберігати працездатність при збоях окремих компонентів, включаючи механізми автоматичного відновлення, реплікації та балансування навантаження. Ця характеристика є визначальною для критичних сервісів, де непередбачені відмови можуть спричинити значні фінансові або репутаційні втрати [11].

Масштабованість системи є показником ефективності, що визначає здатність системи адаптуватися до змінного навантаження без втрати продуктивності і стабільності роботи.

На рис. 3 показано співвідношення ключових метрик роботи системи, які забезпечують високий рівень доступності, надійності та відгуку на зміни в умовах функціонування, що структуроване на основі аналізу впливу ключових метрик роботи системи на інші її характеристики.

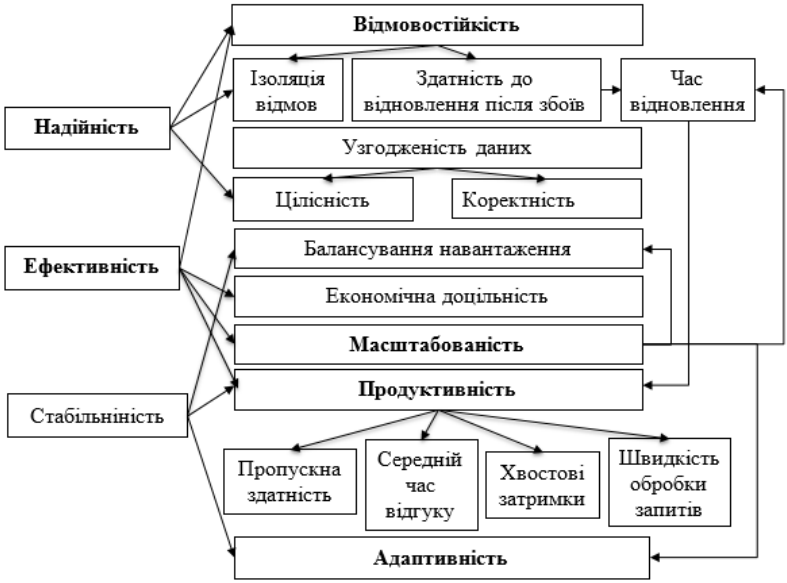

Рис.3. Співвідношення метрик роботи системи

Сформована на основі аналізу метрик ефективності структура є підґрунтям для гіпотези, яка полягає в тому, що механізмом забезпечення

комплексної результативності керування ресурсами є масштабованість, оскільки від цього показника безпосередньо залежать час відновлення роботи системі після збоїв, балансування навантаження і адаптивність, а залежність продуктивності системи від часу відновлення після збоїв виявляє опосередкований вплив масштабованості на продуктивність системи.

Принципи масштабування високонавантажених розподілених систем поділяються на вертикальне і горизонтальне (рис. 4).

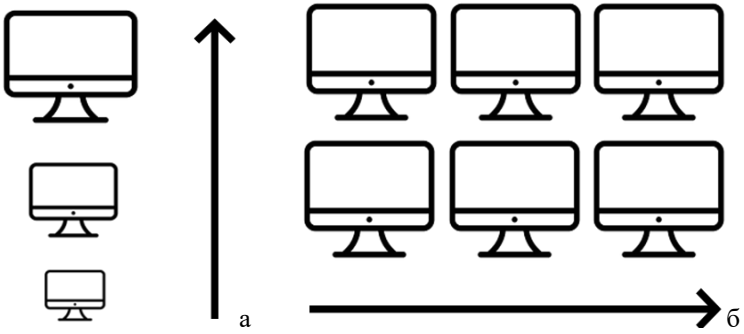

Рис.4. Горизонтальне (а) і вертикальне (б) масштабування

Вертикальне масштабування передбачає збільшення обчислювальної потужності окремого вузла шляхом додавання процесорних ядер, оперативної пам'яті чи інших ресурсів. Ефективність цього підходу в умовах обмежених обчислювальних середовищ, таких як монолітні додатки і бази даних, що потребують збереження цілісності процесів на одному сервері досліджено в [12]. Перевагою вертикального масштабування є зменшення накладних витрат на координацію вузлів, оскільки всі ресурси зосереджені в межах однієї фізичної чи віртуальної машини. Проте вертикальне масштабування часто обмежується ресурсами сервера і витратами на модернізацію обладнання.

Горизонтальне масштабування передбачає додавання обчислювальних вузлів до системи, що забезпечує її лінійне розширення та підвищення пропускної здатності без змін у конфігурації окремих серверів [13]. Такий підхід широко застосовується в розподілених мікросервісних архітектурах, оскільки дозволяє досягати високої відмовостійкості та уникати єдиної точки відмови. Проте, ефективність горизонтального масштабування значно залежить від реалізації балансування навантаження та міжвузлової комунікації, що може створювати додаткові затримки у великих кластерах.

Сучасні високонавантажені системи мають комбіноване масштабування, що дозволяє реалізовувати комбіновані стратегії

управління ресурсами. Таким чином актуальною є задача розробки моделей і методів коригування моменту навантаження, що і буде предметом подальших досліджень.

Висновки. У результаті проведеного дослідження встановлено, що основною спільною рисою високонавантажених систем є необхідність обробляти великі об'єми запитів у реальному часі, забезпечуючи високий рівень доступності, надійності та відгуку на зміни в умовах функціонування. При цьому оптимізація витрат у хмарних обчисленнях і розподілених середовищах є ключовим завданням, особливо з урахуванням динамічного характеру робочих навантажень і необхідності автоматизованого керування ресурсами. Саме тому перед впровадженням власного рішення необхідно проводити аналіз метрик для чіткого розуміння роботи системи під навантаженням і визначення її слабких місць. Це надає змогу оптимально налаштувати масштабування, забезпечити потрібний рівень відмовостійкості і контролювати витрати, гарантуючи стабільний користувацький досвід та відповідність вимогам безпеки і економічної ефективності.

Список використаної літератури

1. Datareportal. Digital 2022: Global Overview Report January 2022. URL: https://datareportal.com/reports/digital-2022-global-overview-report

2. Sandvine. Global Internet Phenomena Report (GIPR) January 2022. Sandvine Inc. URL: https://www.sandvine.com/hubfs/Sandvine_Redesign 2019/Downloads/2022/Phenomena%20Reports/GIPR%202022/Sandvine%20GIPR%20January%202022.pdf

3. Cisco. (2022). Global Networking Trends Report. URL: https://storage.eventcheckin.co.kr/cisco/2023/CXO_symposium/data/2022_Global_Networking_Trend_Report_eng.pdf

4. Weiner M., Xu Y., "Performance Metrics for Distributed Systems", ACM Computing Surveys, 2022, Vol. 54, No. 7, P. 1–29.

5. Dean J., Ghemawat S., "MapReduce: Simplified Data Processing on Large Clusters", Communications of the ACM, 2008, Vol. 51, No. 1, P. 107–113. DOI: https://doi.org/10.1145/1327452.1327492

6. Leis V., Boncz P., "Query Processing for Distributed Databases", IEEE Transactions on Knowledge and Data Engineering, 2021, Vol. 33, No. 5, P. 1784–1802. DOI: https://doi.org/10.1109/TKDE.2020.3036432

7. Drepper U., "The Cost of Latency in Distributed Systems", USENIX; login, 2018, Vol. 43, No. 2, P. 3–10.

8. Abadi D., "Consistency Tradeoffs in Modern Distributed Database Systems", IEEE Data Engineering Bulletin, 2020, Vol. 43, No. 2, P. 22–30. URL: https://sites.computer.org/debull/A20june/2-Abadi.pdf

9. Rao J., "Tail Latency in Distributed Systems", IEEE Internet Computing, 2019, Vol. 23, No. 2, P. 68–77. DOI: https://doi.org/10.1109/MIC.2019.2899036

10. Fox A., Patterson D., "Recovery-Oriented Computing: A New Research Agenda for Highly Available Systems", ACM Transactions on Internet Technology, 2021, Vol. 19, No. 4, P. 1–27.
11. Kanev S., "Profiling Tail Latencies in Data Centers", ACM Symposium on Cloud Computing, 2021, P. 103–116.
12. Papageorgiou G., "Ensuring Reliability in Distributed Systems: A Survey", Future Generation Computer Systems, 2022, Vol. 129, P. 102–124.
13. Armbrust M., "Scalability of Cloud-Based Workloads", USENIX Annual Technical Conference, 2020, P. 245–258.

References

1. Datareportal. Digital 2022: Global Overview Report – January 2022. URL: https://datareportal.com/reports/digital-2022-global-overview-report
2. Sandvine. Global Internet Phenomena Report (GIPR) – January 2022. Sandvine Inc. URL: https://www.sandvine.com/hubfs/Sandvine_Redesign 2019/Downloads/2022/Phenomena%20Reports/GIPR%202022/Sandvine%20GIPR%20January%202022.pdf
3. Cisco. (2022). Global Networking Trends Report. URL: https://storage.eventcheckin.co.kr/cisco/2023/CXO_symposium/data/2022_Global_Networking_Trend_Report_eng.pdf
4. Weiner, M., & Xu, Y. (2022). Performance metrics for distributed systems. *ACM Computing Surveys*, 54(7), 1–29.
5. Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1), 107–113.
6. Leis, V., & Boncz, P. (2021). Query processing for distributed databases. *IEEE Transactions on Knowledge and Data Engineering*, 33(5), 1784–1802, DOI: https://doi.org/10.1109/TKDE.2020.3036432
7. Drepper, U. (2018). The cost of latency in distributed systems. *USENIX; login*, 43(2), 3–10.
8. Abadi, D. (2020). Consistency tradeoffs in modern distributed database systems. *IEEE Data Engineering Bulletin*, 43(2), 22–30.
9. Rao, J. (2019). Tail latency in distributed systems. *IEEE Internet Computing*, 23(2), 68–77. DOI: https://doi.org/10.1109/MIC.2019.2899036
10. Fox, A., & Patterson, D. (2021). Recovery-oriented computing: A new research agenda for highly available systems. *ACM Transactions on Internet Technology*, 19(4), 1–27. DOI: https://doi.org/10.1145/3376907
11. Kanev, S. (2021). Profiling tail latencies in data centers. *In Proceedings of the ACM Symposium on Cloud Computing*, pp. 103–116,
12. Papageorgiou, G. (2022). Ensuring reliability in distributed systems: A survey. *Future Generation Computer Systems*, 129, 102–124.
13. Armbrust, M. (2020). Scalability of cloud-based workloads. *In Proceedings of the USENIX Annual Technical Conference*, pp. 245–258.

Principles and methods for automatic scaling of high-load systems

This paper investigates a wide spectrum of use cases and the main approaches applied to Internet-traffic processing by industry giants such as Netflix, YouTube, Disney+, TikTok, Facebook, Xbox Live, PlayStation Downloads, and Amazon Prime. It has been demonstrated that the defining characteristic of high-load systems is the need for automated resource management to process large volumes of real-time requests while ensuring high availability, reliability and stability amid dynamically changing workloads. The impact of key performance metrics on other functional characteristics has been analyzed. Based on these analyze, the system's key metrics set that guarantees system's responsiveness under unpredictable conditions has been systematized and structured. Findings has indicated that system throughput depends on adaptive load balancing, whereas scalability determines the system's capacity to adapt. Thus, efficient resource management and adaptive scaling algorithms makes it possible to reduce costs without losing performance. It has also been shown that the efficiency and resilience of high-load systems significantly depend on architectural solutions. Moreover, vertical scaling of the system enables increased performance of individual system nodes, while horizontal scaling provides better fault tolerance. It has been established that the main challenge in achieving economic efficiency lies in balancing scalability, performance and fault tolerance, while combined scaling strategies can help align these objectives. The relevance of developing models and methods for optimizing load adjustments timing has been highlighted, and directions for further research are outlined. The results these investigation may be used in the future to develop adaptive models and management methods of high-load distributed systems resources that are able to operate effectively under conditions of dynamic unpredictable loads and increasing performance requirements..

Keywords: adaptability, dynamic scaling, cost efficiency, information system, information technology, high performance, resource management, software.

Посилання на статтю:

APA: Buhrov Anatolii, Terenchuk Svitlana (2023). Principles and methods for automatic scaling of high-load systems. *Shliakhy pidvyshchennia efektyvnosti budivnytstva v umovakh formuvannia rynkovykh vidnosyn, 52(3), 217-226.*

ДСТУ: Бугров А.А., Теренчук С.А. Принципи і методи автоматичного масштабування високонавантажених систем. *Шляхи підвищення ефективності будівництва*. 2023. № 52(3). С. 217-226.